

# Federico Abeldaño

Full Stack Developer with experience in Web Development,  
Blockchain Integration, and Database Management

Get to Know

## About Me



### Experience

As a freelancer developer, I have applied my skills in web development, blockchain integration, database management, API creation, deployment of applications in Docker containers, and utilization of styling and design libraries. I have worked on both frontend and backend projects, utilizing a variety of technologies and tools to deliver effective and high-quality solutions to my clients.



### Projects

- frontend-ispo
- dashboards-fe
- dashboards-be
- snapshot
- chatPDF
- CRUD-AWS
- CRUD
- ReservasColas
- PantallaMultiRedes
- CurriculumVitae
- FrontPanel
- FrontDeploySmartContract
- GasFeDev-PortfolioWebSite
- api-GraphQL-ApolloServer
- Squarespace Templates

**Summary:** An Argentine national currently residing in Malargüe, Mendoza (Argentina). Freelance developer with skills in web development, blockchain integration, database management, API creation, and deploying applications using Docker containers. Proficient in both frontend and backend development, leveraging a wide range of technologies and tools. Committed to professional growth and continuous learning, taking my first steps in the world of programming in 2021. Hold a degree in Industrial Engineering from the National Technological University (2017). Ever since I entered the world of Programming, I felt immense satisfaction with each step I took and with every knowledge I acquired. That's why, regardless of the learning and experience gained through my Degree, my short-term or medium-term goal is to earn one hundred percent of my income as a Developer, and above all, to work the rest of my days in something that I am passionate about and brings me happiness.

## My Experience

### Frontend Development

📌 <b>HTML</b> Intermediate	📌 <b>CSS</b> Intermediate	📌 <b>JavaScript</b> Intermediate
📌 <b>TypeScript</b> Intermediate	📌 <b>React</b> Intermediate	📌 <b>Bootstrap</b> Intermediate
📌 <b>Material-UI</b> Intermediate	📌 <b>Ant Design</b> Intermediate	📌 <b>Recharts</b> Basic
📌 <b>Styled Components</b> Intermediate	📌 <b>Next.js</b> Basic	

### Backend Development

📌 <b>JavaScript</b> Intermediate	📌 <b>TypeScript</b> Intermediate	📌 <b>Node JS</b> Intermediate
📌 <b>Express</b> Intermediate	📌 <b>Axios</b> Intermediate	📌 <b>MoongoDB</b> Intermediate
📌 <b>Dgraph</b> Intermediate	📌 <b>AWS</b> Intermediate	📌 <b>Formik</b> Intermediate
📌 <b>JWT</b> Intermediate	📌 <b>Docker</b> Intermediate	📌 <b>PHP</b> Basic

What I know

## Competences

### Frontend Development

- ✓ Languages: JavaScript, TypeScript.
- ✓ Libraries and Frameworks: React, Material-UI, Ant Design, Bootstrap.
- ✓ Styling and Design: CSS, Emotion, Styled Components.
- ✓ Data Visualization: Recharts.
- ✓ Using Next.js as a React framework.

### Other Tools and Technologies

- ✓ Date and Time Manipulation: Moment.js, Day.js.
- ✓ PDF File Manipulation: pdf-lib, react-pdf.
- ✓ Routing and Request Handling: React Router.
- ✓ State Management: React Context, Redux.
- ✓ AWS: S3, DynamoDB, Amazon API Gateway.
- ✓ Render.com: Application deployment.

### Backend Development

- ✓ Languages: JavaScript, TypeScript.
- ✓ Frameworks: Express.
- ✓ Databases: MongoDB, Dgraph.
- ✓ Communication with External Services: Axios.
- ✓ Form Validation: Formik, Yup.
- ✓ Authentication and Authorization: JWT.

### Blockchain Interaction

- ✓ Ethereum: Knowledge in interacting with the Ethereum network and utilizing smart contracts through the Ethers library.
- ✓ Ethereum Networks: Ability to read contract states, send transactions, and make specific function calls, demonstrating your understanding and practical application of blockchain technology.

### API Development

- ✓ Using Express as a framework for creating RESTful APIs.
- ✓ Interacting with GraphQL API: Apollo Server, GraphQL.
- ✓ Interacting with databases like Dgraph.
- ✓ Communication with External Services: Axios.
- ✓ Making HTTP requests with Axios

### Application Deployment

- ✓ Docker Containers: Dockerfile, image creation.
- ✓ Container Orchestration: Docker Compose.
- ✓ Docker Network Configuration.
- ✓ Using Nginx for serving applications and configuring proxies.



*Github*

Website



## Github

Website



*Github*

Website



*For both projects, I created and configured Docker containers in a local environment. Docker was used to implement a container architecture. I also took advantage of Docker's ability to create networks and establish communication between containers. I configured a Docker network and linked the containers using that network, allowing smooth communication between them.*

Select the amount you want to stake and participate for rewards

Network  
Ethereum Mainnet

I want to  
Stake

Select Token  
stETH

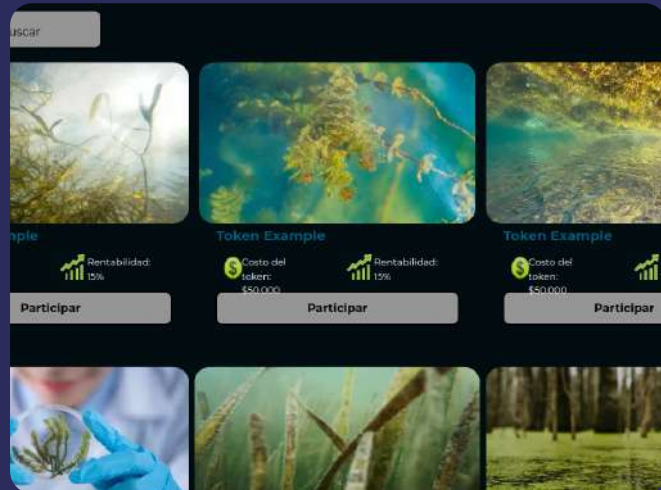
Your Balance  
0.412915407932375257 stETH

Amount you want to stake  
0.1

Total staked balance: 0.000000000000000000

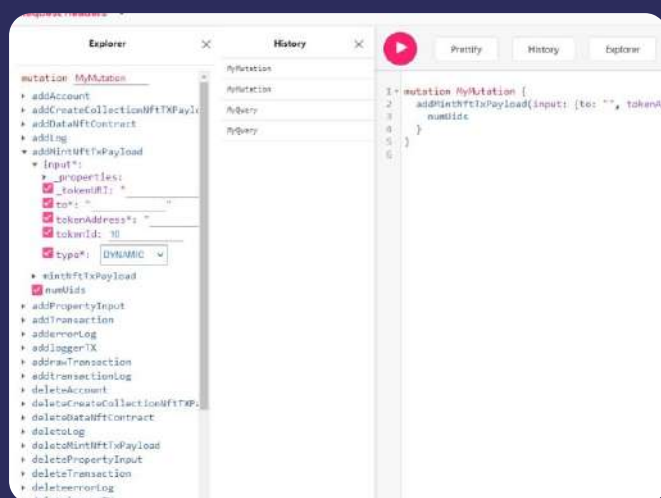
## frontend-ispo

Project developed in JavaScript with React, using Bootstrap CSS. The project involves interaction with the Ethereum network and smart contracts using the Ethers library. Additionally, RainbowKit and Wagmi were used. These libraries have been fundamental in achieving an engaging and seamless interaction with the smart contracts.

[Github](#)
[Website](#)


## FrontPanel

Project developing a protocol based on React, using Next.js. For styling management, I used libraries like Emotion and Styled Components. Material-UI was used to design the user interface and visual components.

[Github](#)
[Website](#)


## api-GraphQL-ApolloServer

Project developing an API using TypeScript. For API development, I used Express, and for creating GraphQL servers, Apollo Server was employed—a GraphQL development platform. Dgraph was used to interact with databases, and Axios was utilized for handling HTTP requests and responses. Web3 and web3-providers-http were also used to interact with the Ethereum network and smart contracts.

[Github](#)
[Website](#)

Tab1 Tab2

Código

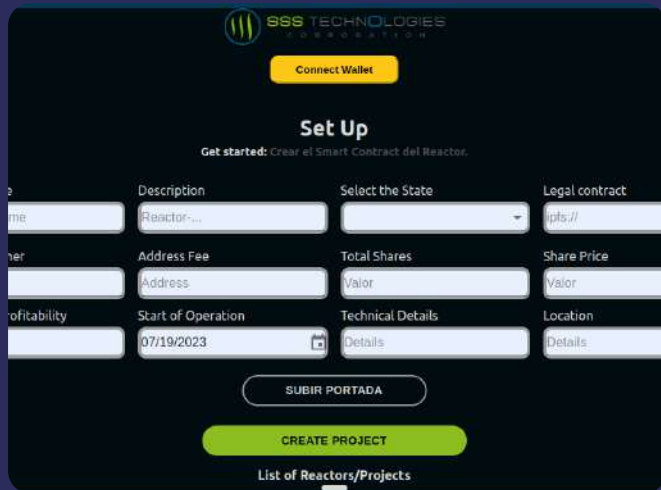
Nombre Item

Descripción

## CRUD-AWS

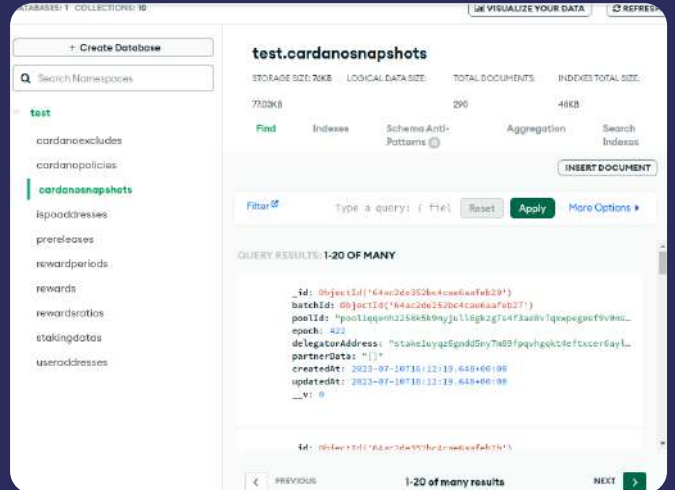
I used React and Bootstrap for the application's design and appearance. React Bootstrap was used for form management and data interaction. I integrated webcam functionality using React Webcam, enabling users to capture images. In this project, data creation, storage, updating, and deletion were achieved through HTTP requests using the GET and PUT methods to an Amazon URL, where management data tables were previously created in DynamoDB, and endpoints were configured through Amazon API Gateway for facilitating communication with the requests.

[Github](#)
[Website](#)



## FrontDeploySmartContract

Project developed in Next.js. For the user interface design and styling, I used the MUI and RainbowKit UI libraries. For state management and interaction with smart contracts on the Ethereum network, I used React Hook Form and Ethers libraries.

[Github](#)
[Website](#)


## snapshot

Project developed on the server-side, where data is processed, interactions with a database are performed, and services and functionalities are provided through an API. TypeScript was used as the programming language. For server development and creating the API, I used Express. Mongoose was employed to interact with MongoDB databases. Axios was used to make HTTP requests to other APIs. Similar to the cases of dashboards-be and dashboards-fe, I created and configured Docker containers for both local development and production environments.

[Github](#)
[Website](#)


## chatPDF

Development of a Web Application using React. For the application's design and style, I used the Material-UI framework. I also used @emotion/react and @emotion/styled to manage styles with CSS in JS. As for communication with cloud services, I utilized the aws-sdk and @aws-sdk/client-s3 libraries to interact with AWS S3 cloud storage service. Axios was used to send requests to the chatGPT API.

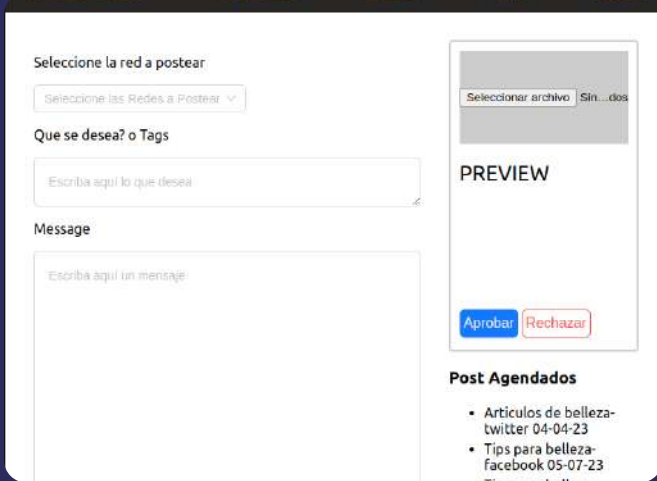
[Github](#)
[Website](#)


## PortfolioWebSite

Project developed in React. For adding style and icons to my website, I used the React Icons library. To send emails from the website, I used the EmailJS and emailjs-com libraries.

[Github](#)
[Website](#)





## PantallaMultiRedes

Project developed in JavaScript with React, utilizing the styling and design capabilities of Ant Design, a user interface framework based on React that provided a wide range of pre-defined components and aesthetic styles, allowing me to build a modern and attractive user interface.

[Github](#)
[Website](#)


## CurriculumVitae

Project developed using a combination of HTML, CSS, and JavaScript along with jQuery and unicons libraries to create an attractive and functional webpage. The HTML document was structured using appropriate tags, configured meta tags for character set and initial page scale. Custom CSS files and an external icon library were linked. Additionally, JavaScript was utilized to provide additional functionality such as DOM manipulation and event handling.

[Github](#)
[Website](#)


## ReservasColas

Project developed in JavaScript with React, featuring interaction with the Ethereum network and smart contracts using the Ethers library, allowing to read the contract's state, send transactions, and make specific function calls. Additionally, to further enhance the user experience and create interfaces rich in animations and transitions, I have utilized RainbowKit and Wagmi. These libraries have been crucial in achieving an engaging and seamless interaction with the smart contracts.

[Github](#)
[Website](#)

## Contact Me



### Web Site

<https://gasfedev-portfoliowebsite.vercel.app/>

[Visit my Web Site](https://gasfedev-portfoliowebsite.vercel.app/)



### Email

[gasfedev@gmail.com](mailto:gasfedev@gmail.com)

[Send a message](mailto:gasfedev@gmail.com)



### Linkedin

<https://www.linkedin.com/>

[Send a message](https://www.linkedin.com/)



### GitHub

<https://github.com/GasFeDev>

[Visit my repositories](https://github.com/GasFeDev)



### Whatsapp

[+542604225858](https://wa.me/542604225858)

[Send a message](https://wa.me/542604225858)